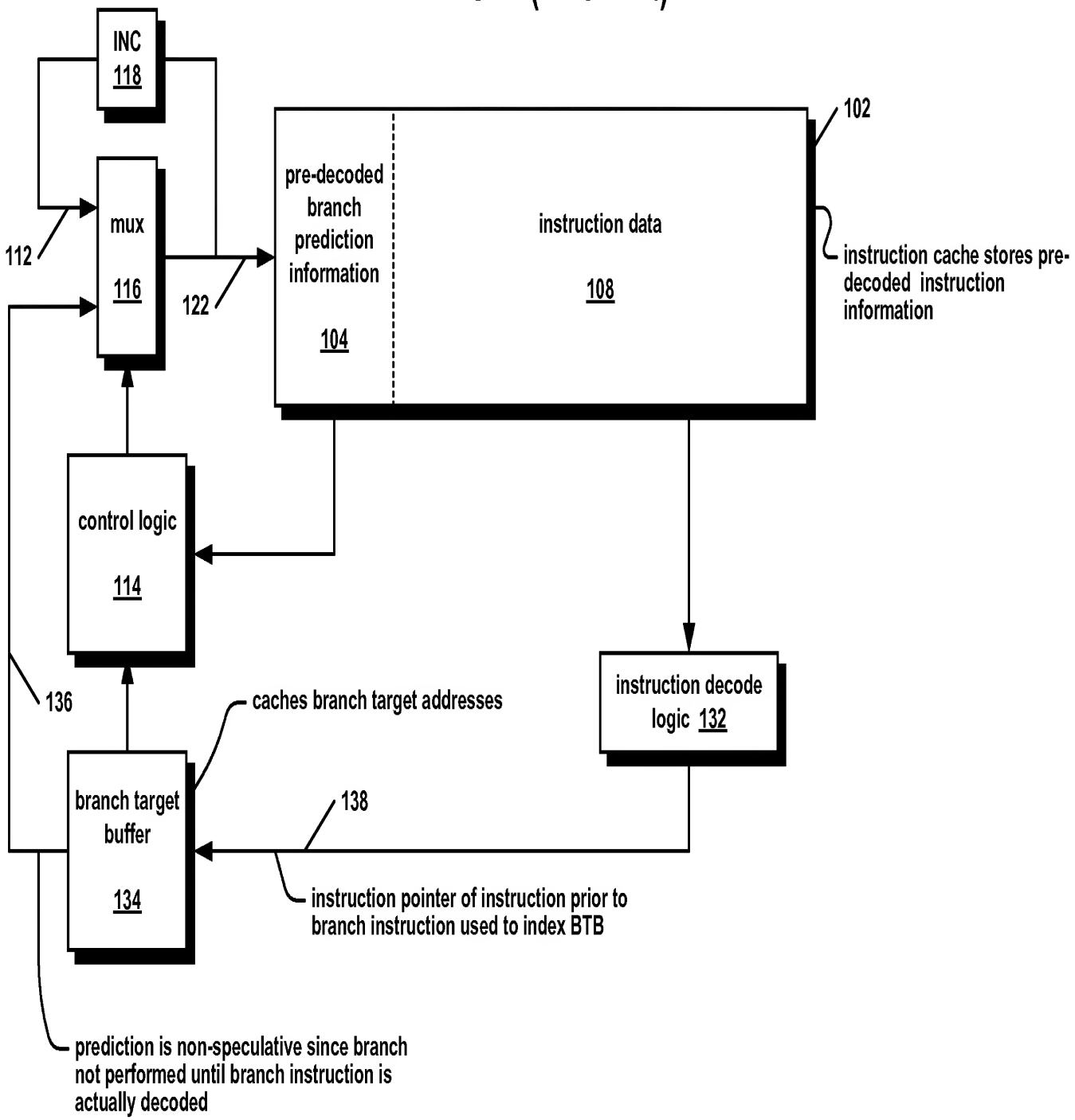
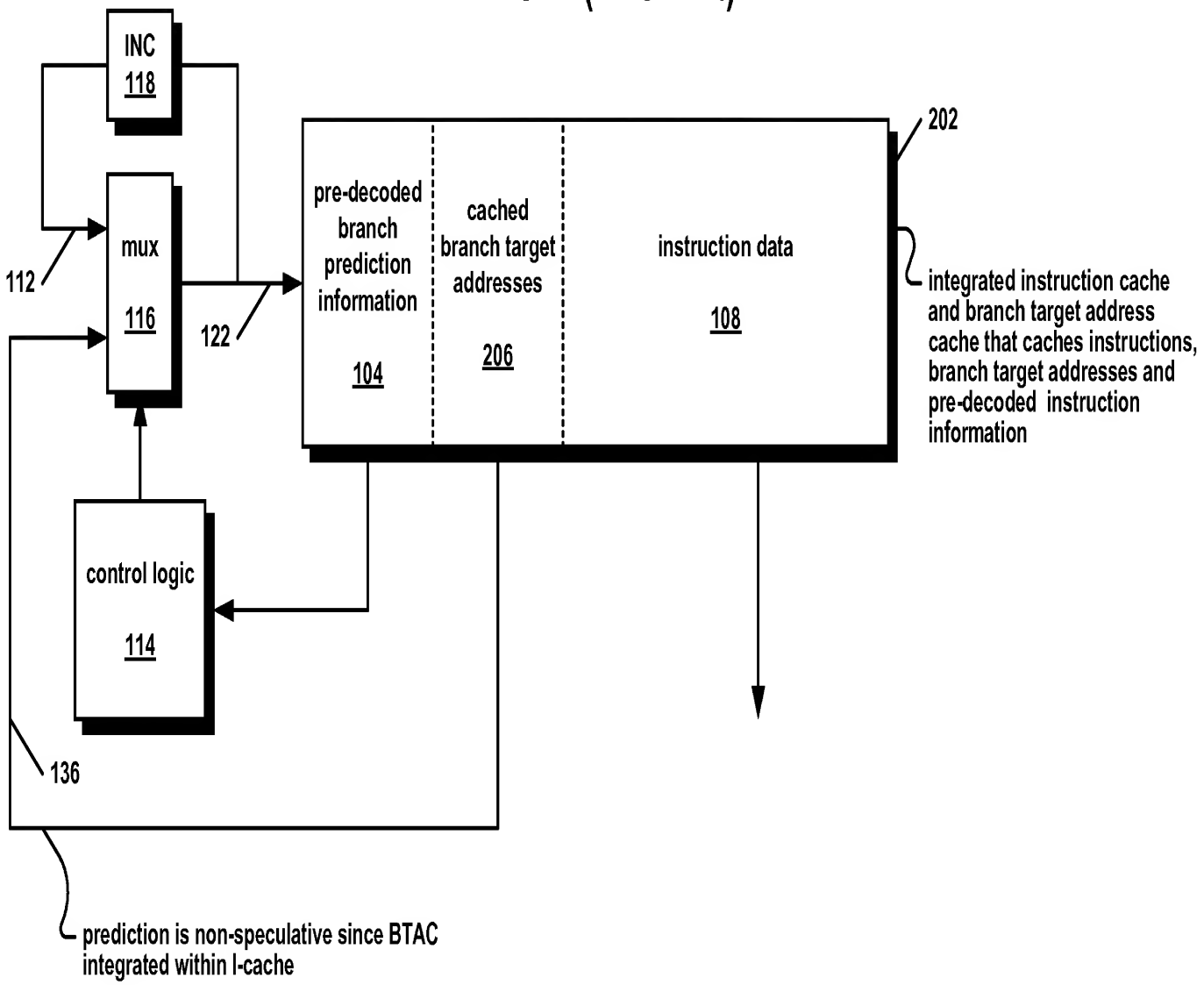
FIG. 1 (Prior Art)



Pentium II, III Branch Target Buffer

100

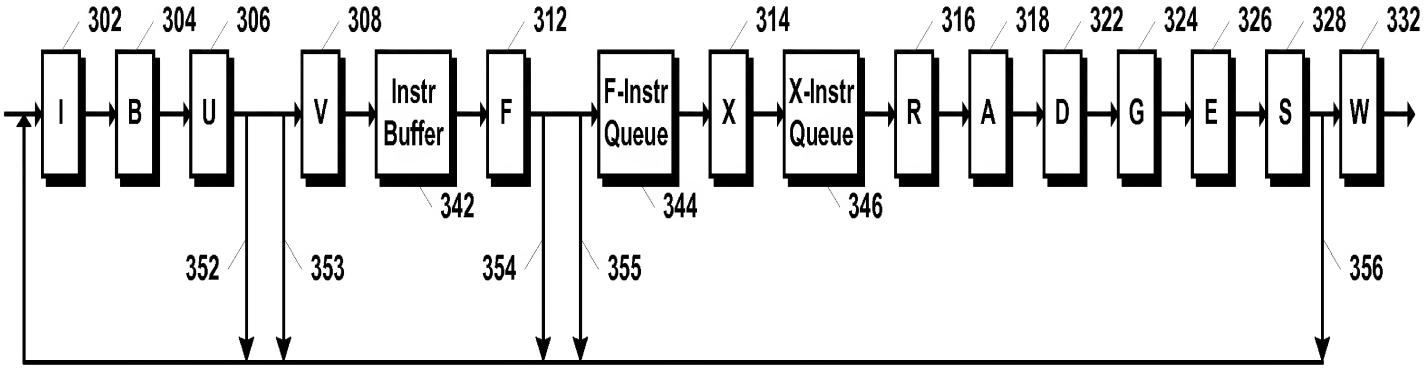
FIG. 2 (Prior Art)



Athlon BTAC Integrated into Instruction Cache

200

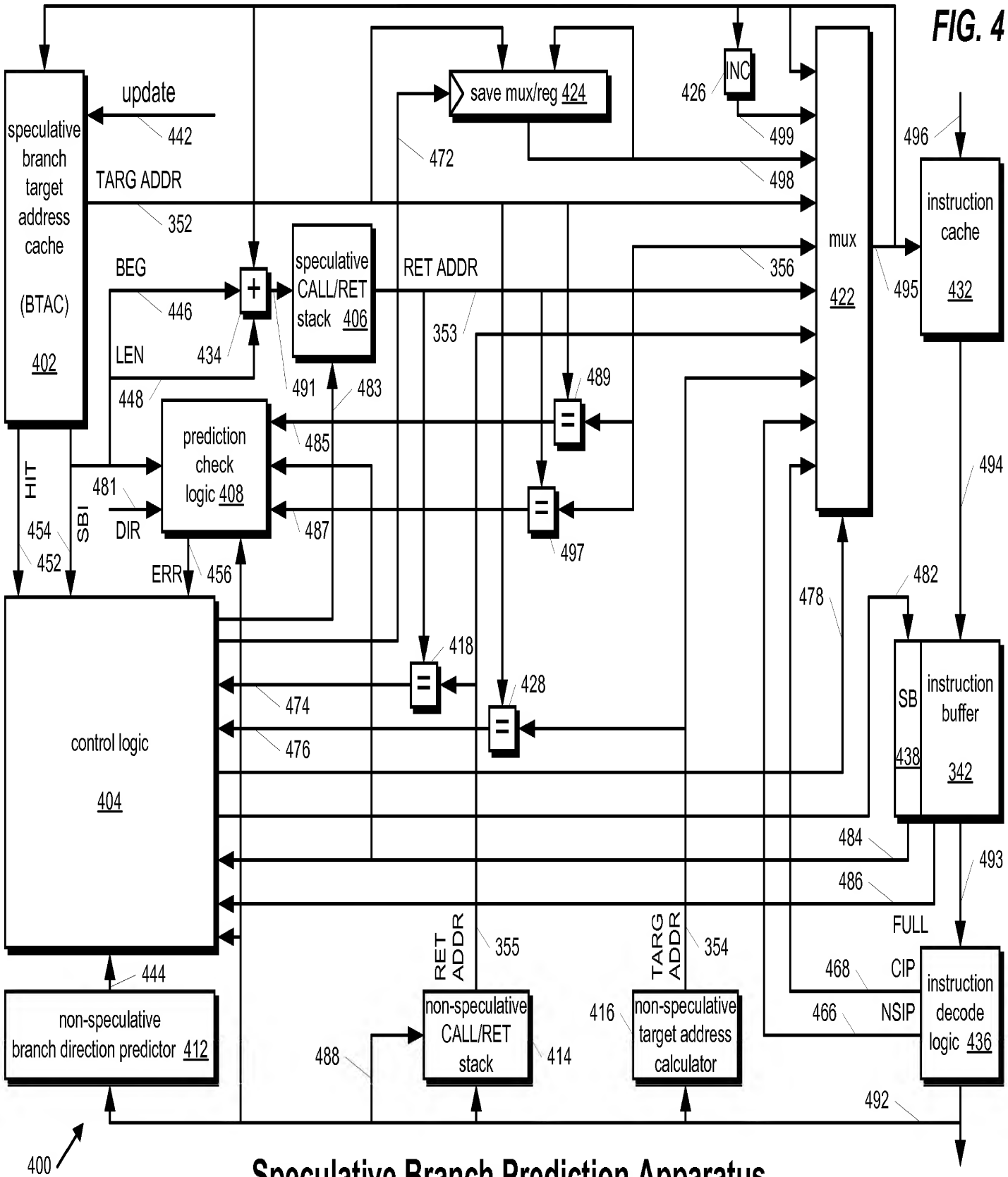
FIG. 3



300

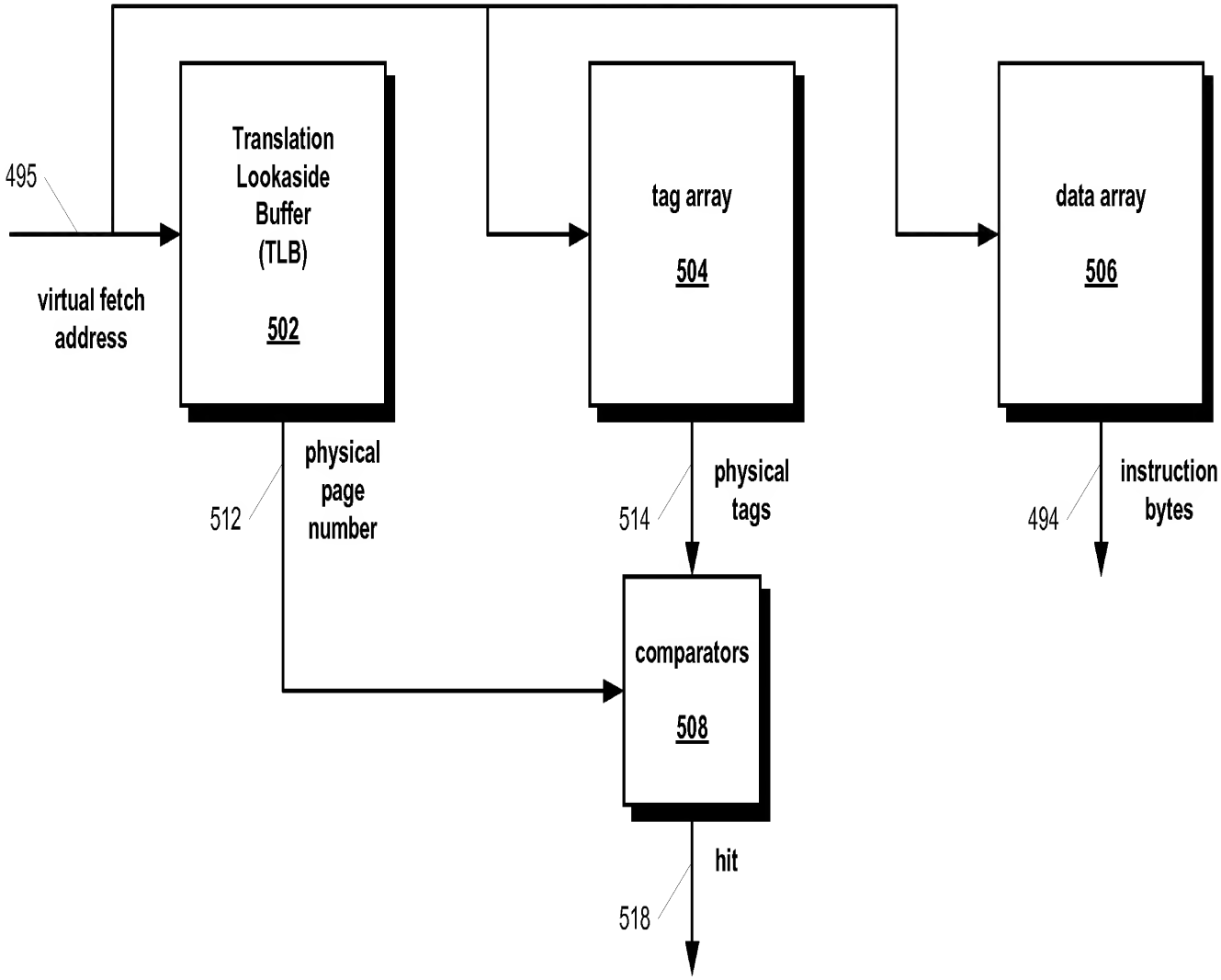
Processor Pipeline Stages

FIG. 4



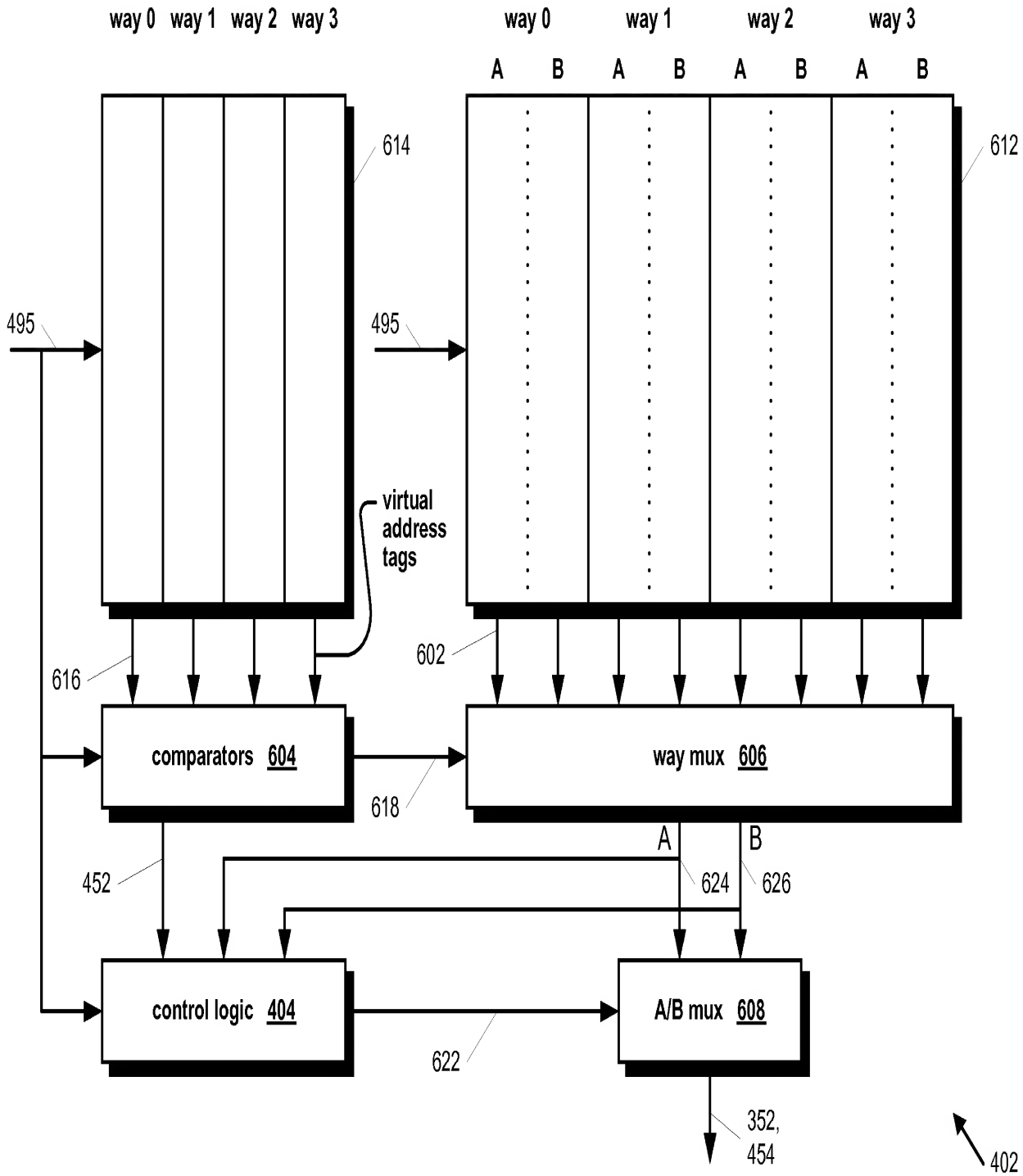
Speculative Branch Prediction Apparatus

FIG. 5



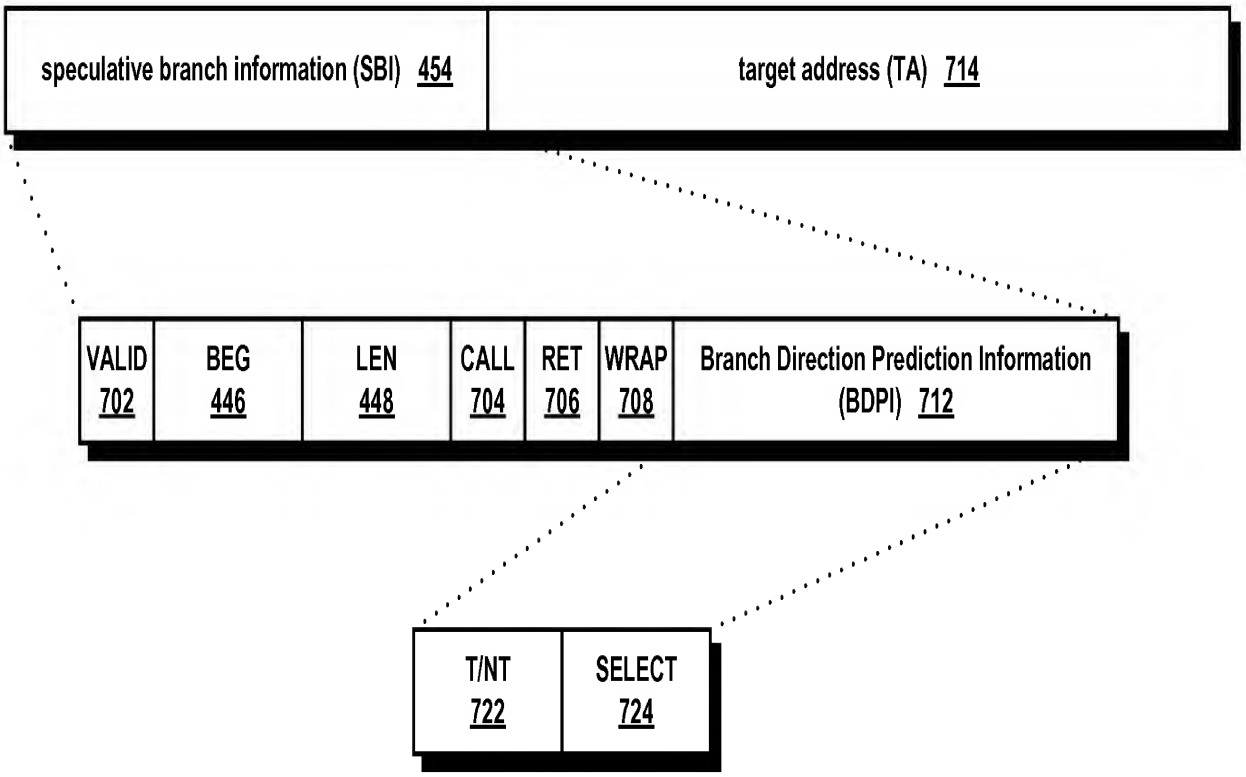
Instruction Cache

FIG. 6


way 0
way 1
way 2
way 3
way 0
way 1
way 2
way 3
A
B
A
B
A
B
A
B
614
612
495
495
virtual address tags
616
602
comparators 604
way mux 606
618
A
B
452
624
626
control logic 404
A/B mux 608
622
352, 454
402

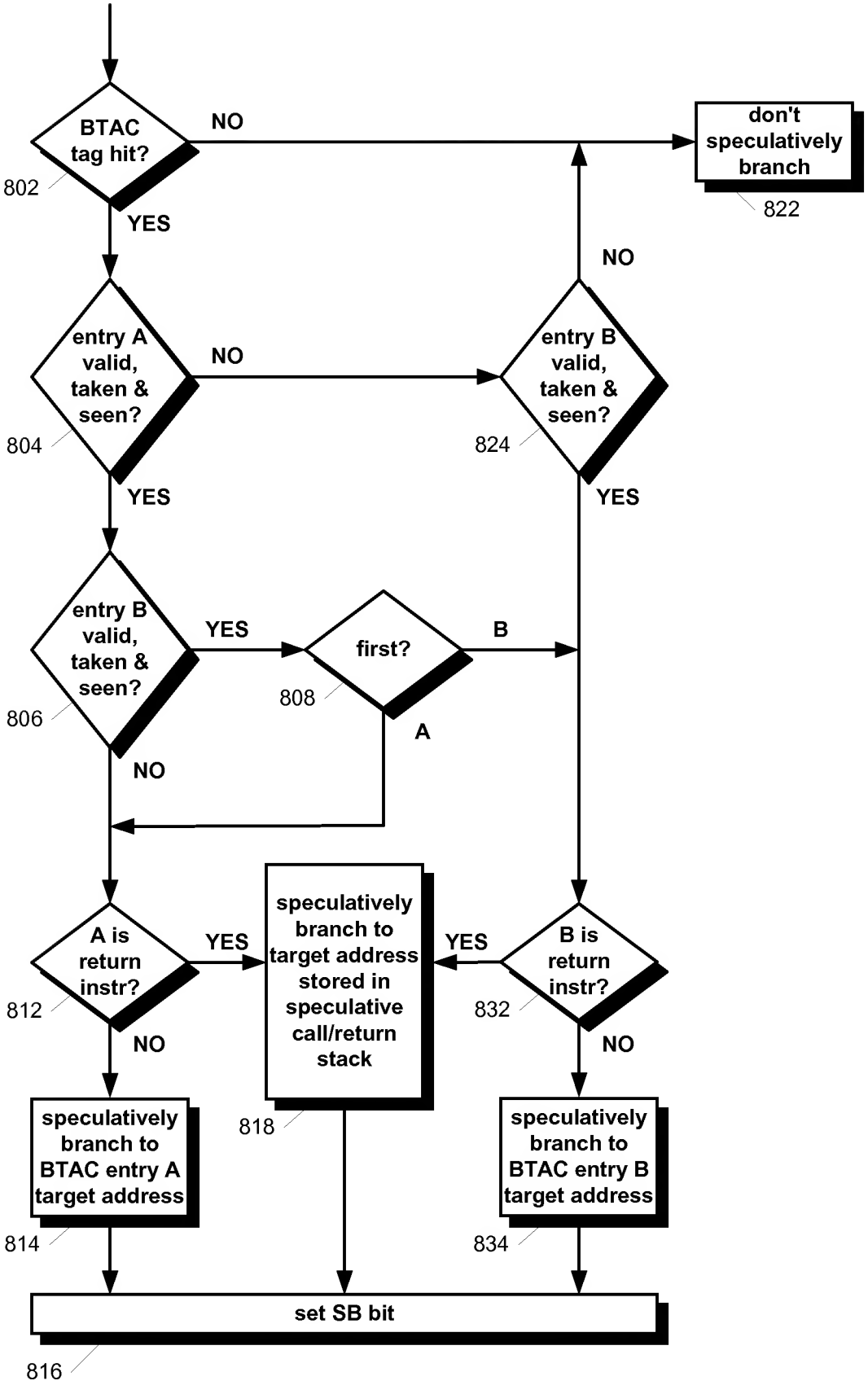
BTAC

FIG. 7



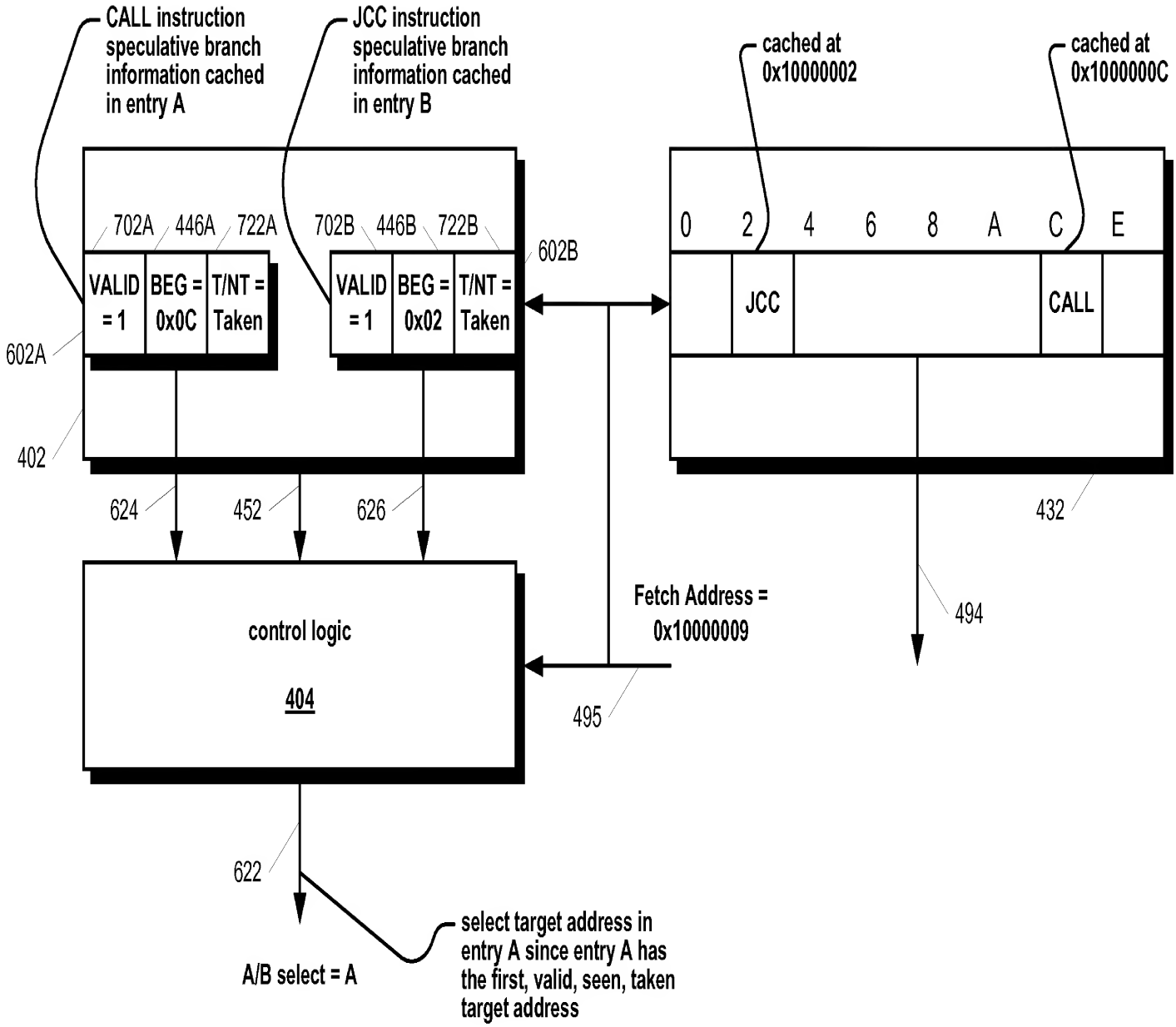
BTAC Entry

FIG. 8



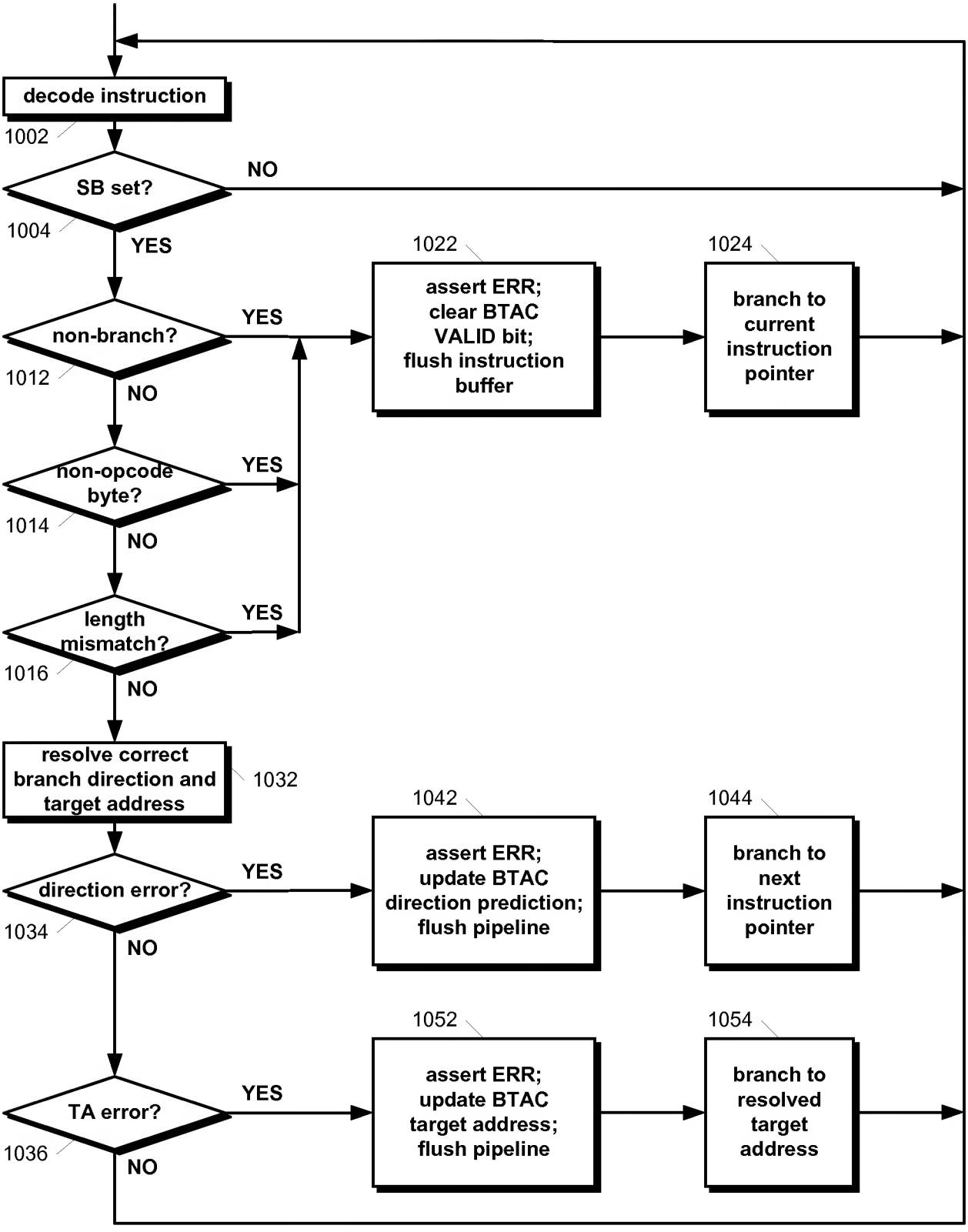
Speculative Branching Operation

FIG. 9



Target Address Selection Example

FIG. 10



Detection and Correction of Speculative Branch Misprediction

FIG. 11

Previous Code Sequence:

0x00000010 JMP 0x00001234
…

Current Code Sequence:

0x00000010 ADD ;address 0x00000010 hits in BTAC generating a TA value of 0x00001234
…

0x00001234 SUB
0x00001236 INC

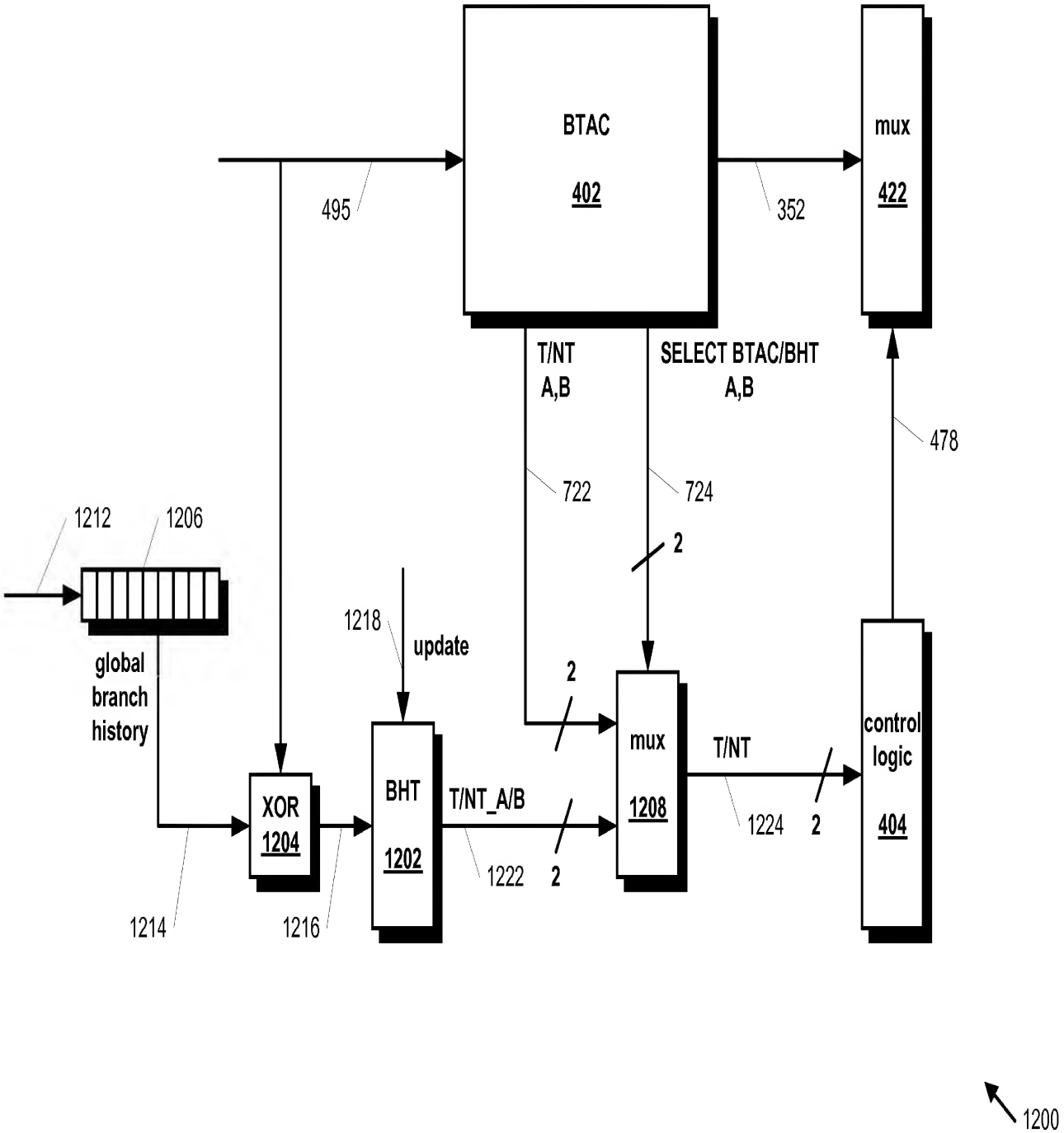
clock →	1	2	3	4	5	6	7
I-stage	ADD	X	X	SUB	INC	X	ADD
B-stage		ADD	X	X	SUB	X	X
U-stage			ADD	X	X	X	X
V-stage				ADD	X	X	X
F-stage					ADD	X	X

Cycle 1 = BTAC and I-cache access cycle
Cycle 4 = speculative branch cycle
Cycle 5 = speculative branch error detection cycle
Cycle 6 = BTAC invalidate cycle
Cycle 7 = speculative branch error correction cycle

1100

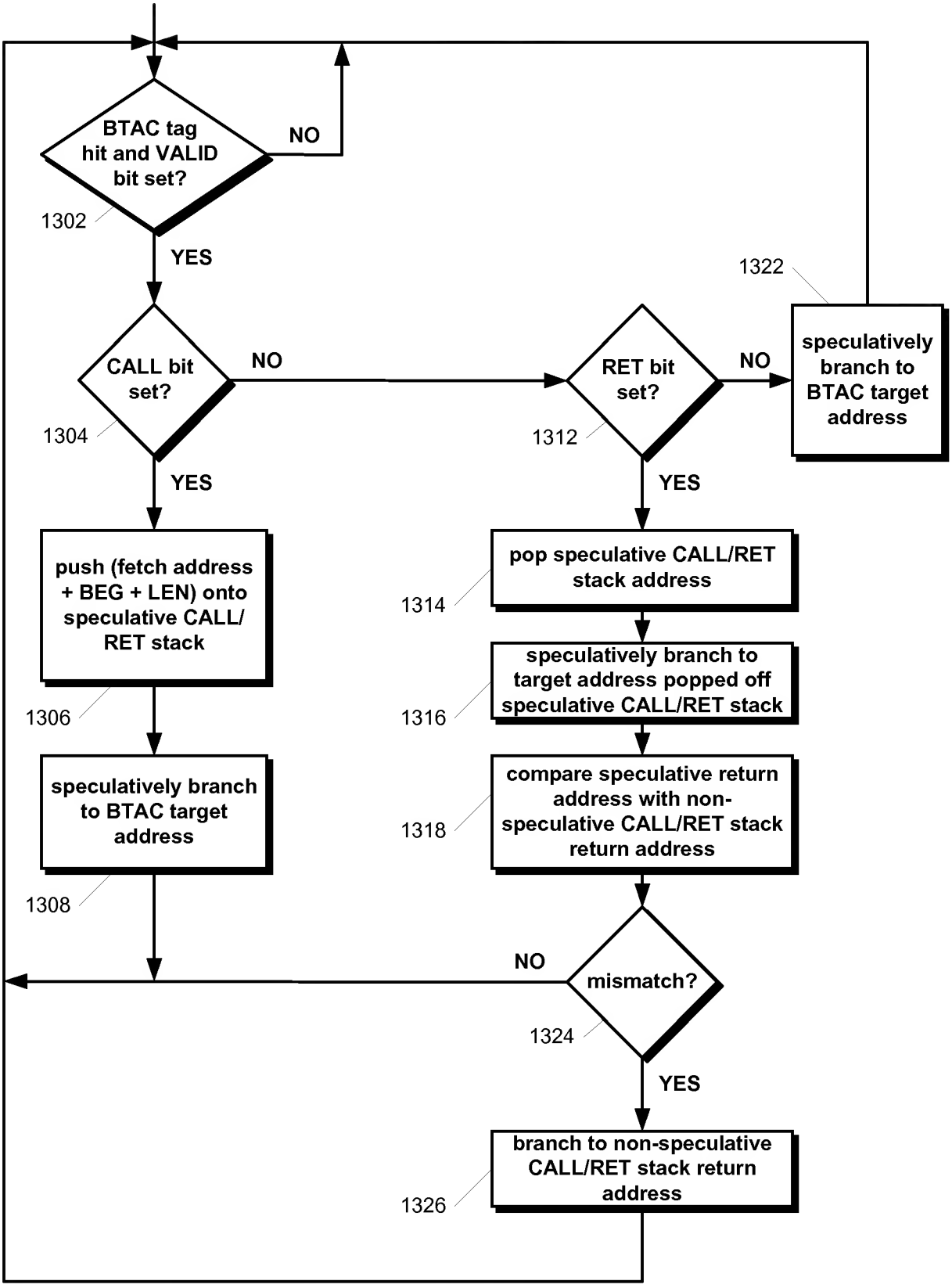
Misprediction Detection and Correction Example

FIG. 12



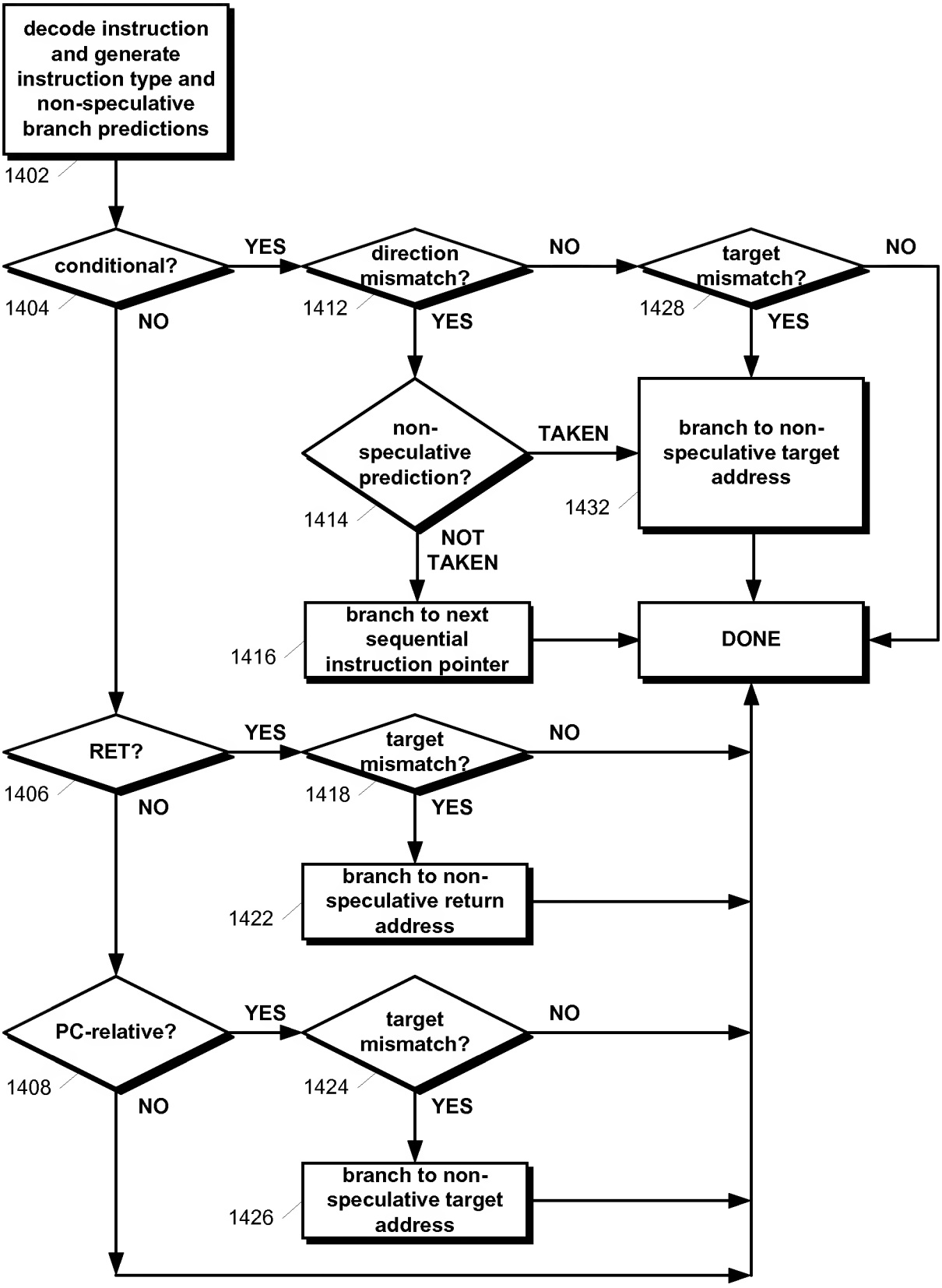
Hybrid Speculative Branch Direction Predictor

FIG. 13



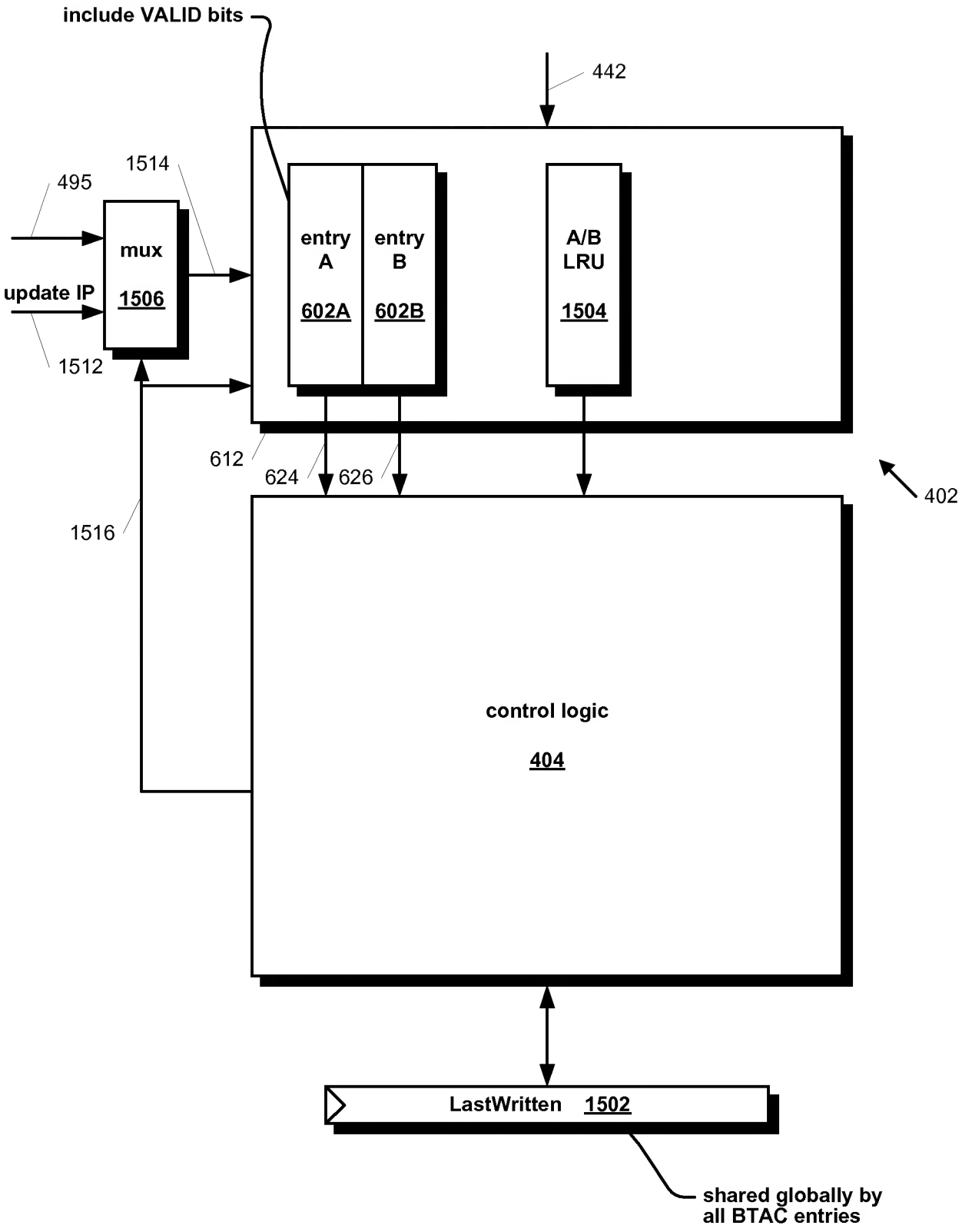
Dual CALL/RET Stack Operation

FIG. 14



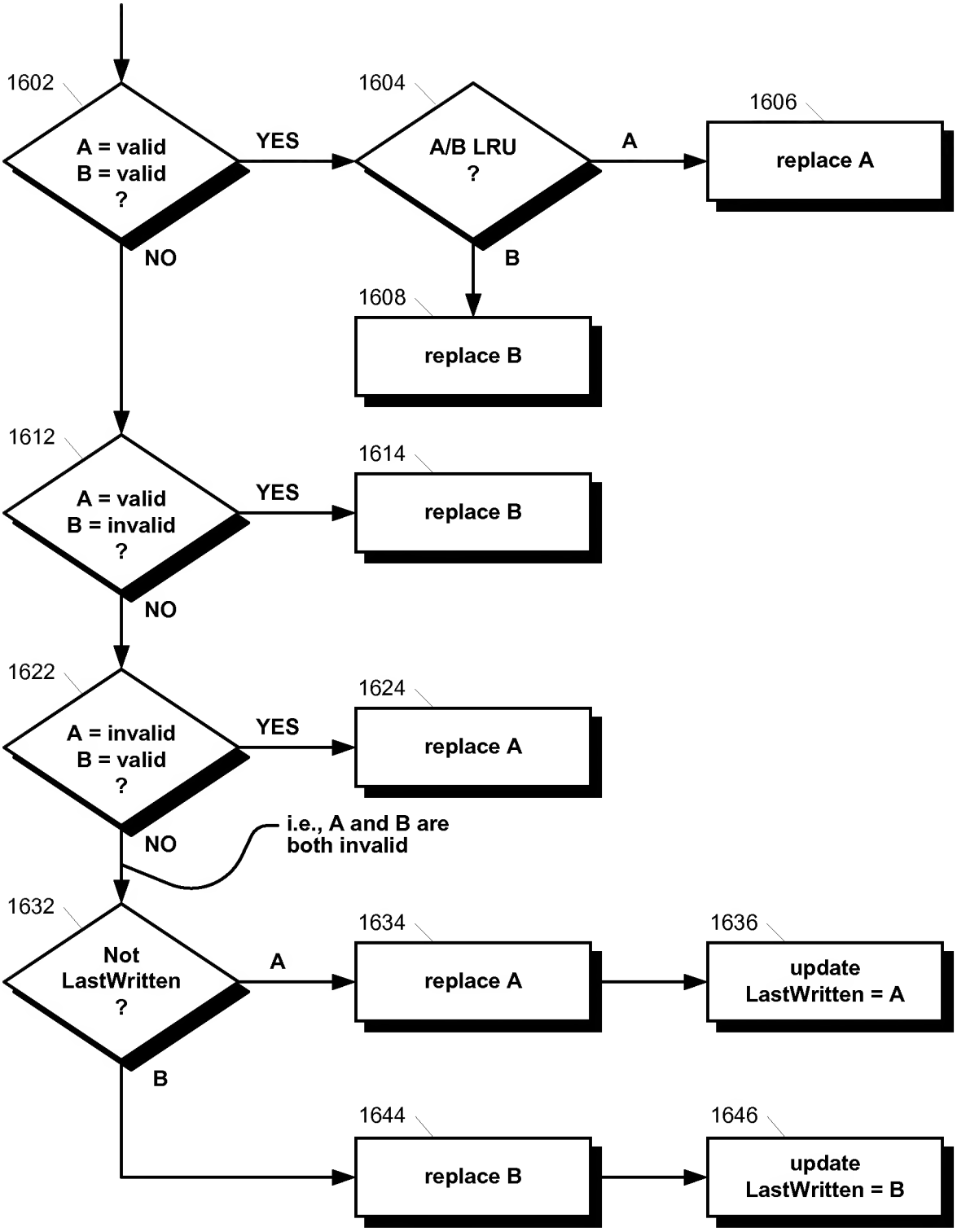
Selective Override of BTAC Prediction Operation

FIG. 15



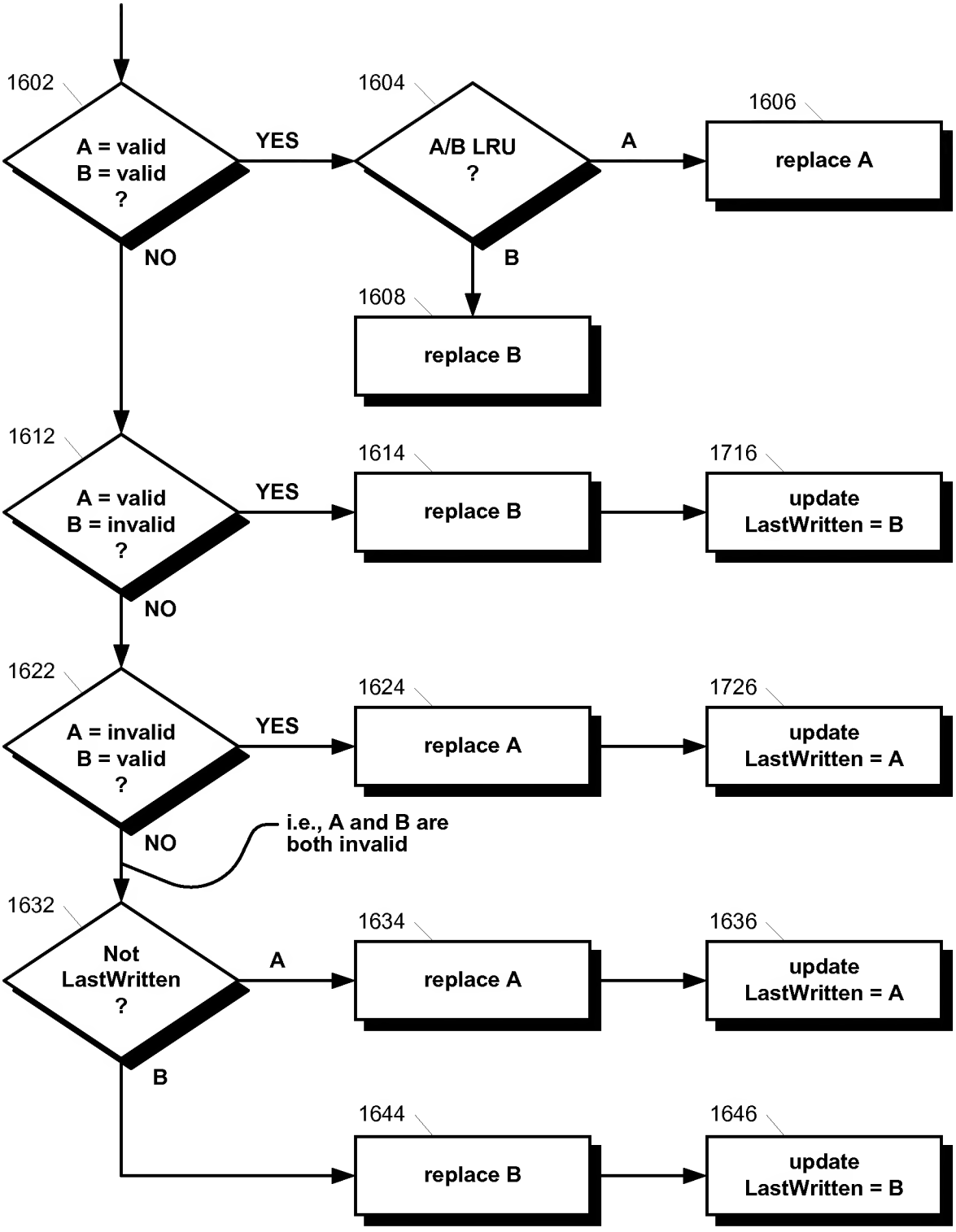
BTAC A/B Replacement Apparatus

FIG. 16



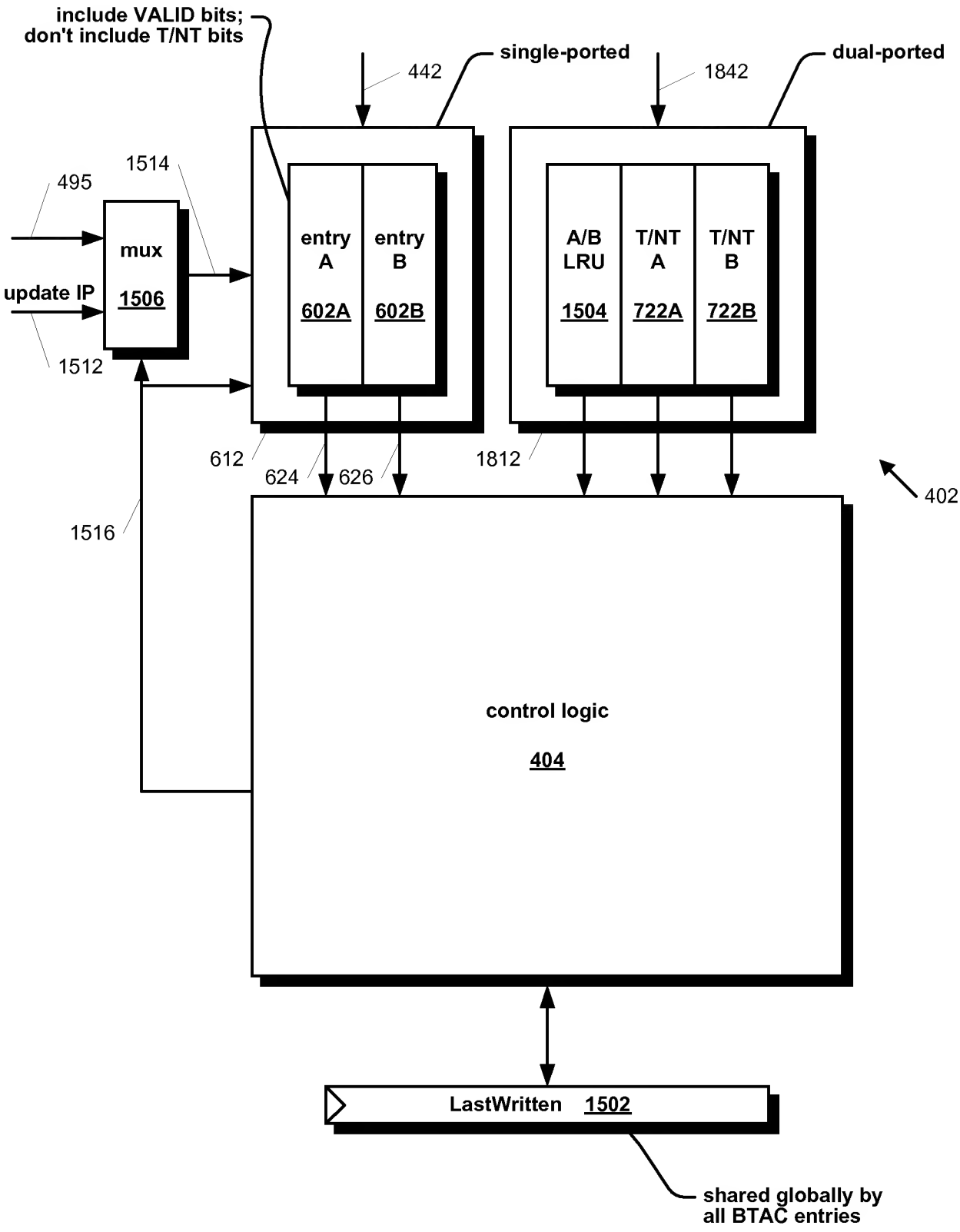
A/B Entry Replacement Method

FIG. 17



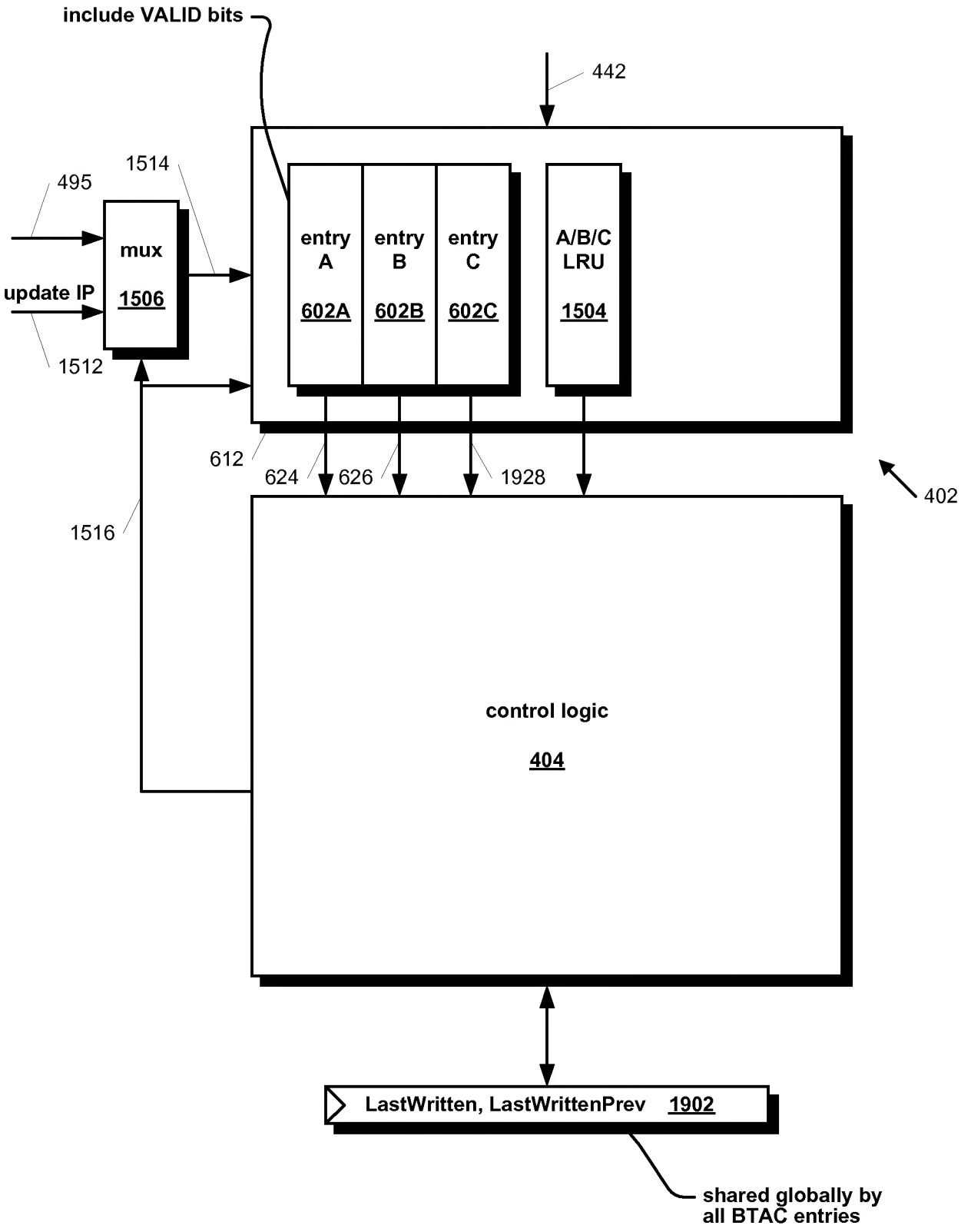
A/B Entry Replacement Method (Alt. Embodiment)

FIG. 18



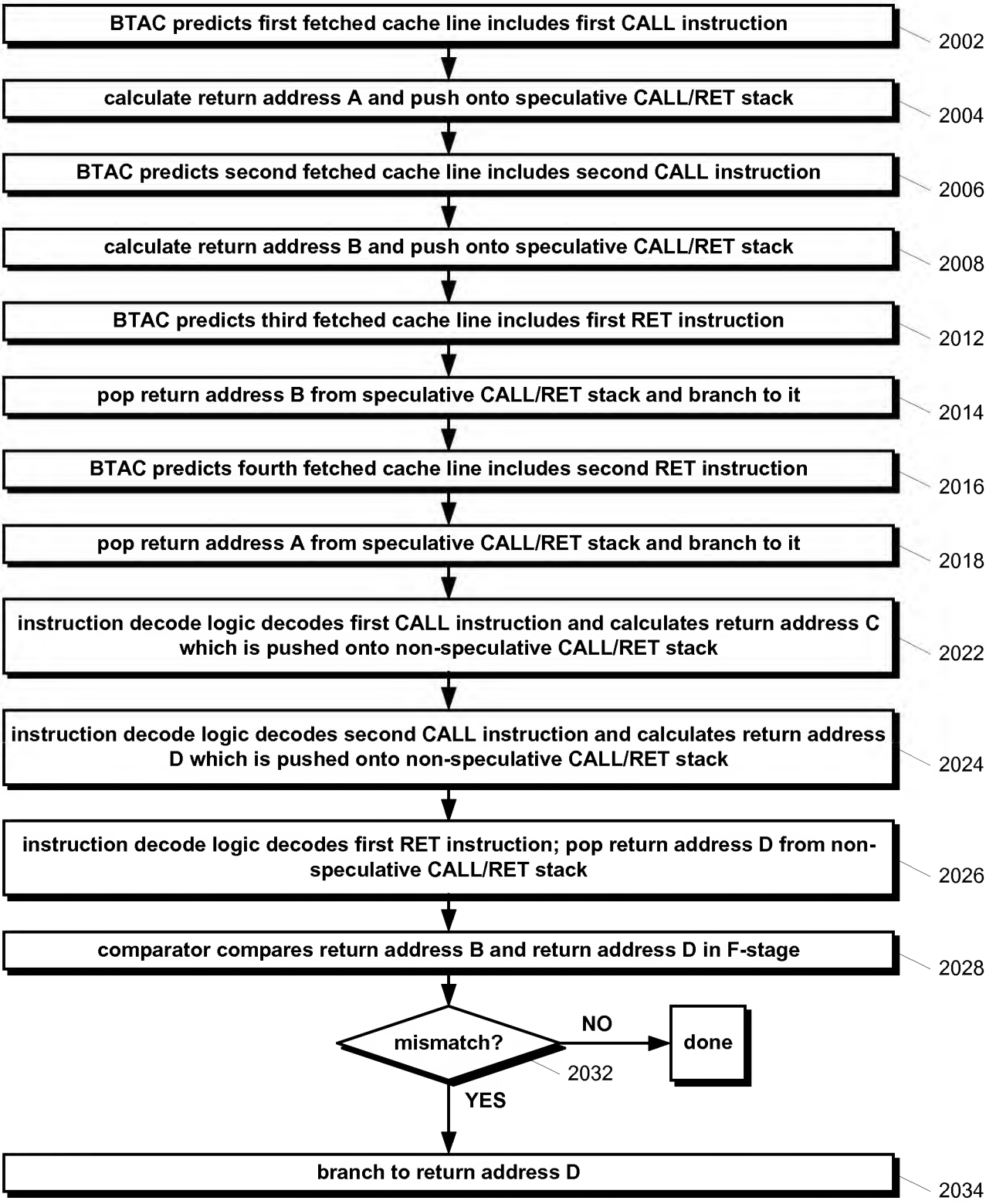
BTAC A/B Replacement Apparatus (Alt. Embodiment)

FIG. 19



BTAC A/B/C Replacement Apparatus

FIG. 20


BTAC predicts first fetched cache line includes first CALL instruction
2002
calculate return address A and push onto speculative CALL/RET stack
2004
BTAC predicts second fetched cache line includes second CALL instruction
2006
calculate return address B and push onto speculative CALL/RET stack
2008
BTAC predicts third fetched cache line includes first RET instruction
2012
pop return address B from speculative CALL/RET stack and branch to it
2014
BTAC predicts fourth fetched cache line includes second RET instruction
2016
pop return address A from speculative CALL/RET stack and branch to it
2018
instruction decode logic decodes first CALL instruction and calculates return address C which is pushed onto non-speculative CALL/RET stack
2022
instruction decode logic decodes second CALL instruction and calculates return address D which is pushed onto non-speculative CALL/RET stack
2024
instruction decode logic decodes first RET instruction; pop return address D from non-speculative CALL/RET stack
2026
comparator compares return address B and return address D in F-stage
2028
mismatch?
NO
done
2032
YES
branch to return address D
2034